

**Amendments to the Claims:**

This listing of claims will replace all prior versions, and listings, of claims in the application:

**Listing of Claims:**

Claims 1-21 (canceled).

Claims 22 - 23 (cancel)

Claim 24 (currently amended): The method of claim ~~[[23]]~~ 31, further comprising:  
storing in a control field associated with the previous instruction a value to indicate that  
the compressed operand should be ignored when executing the previous instruction.

Claim 25 (cancel)

Claim 26 (currently amended): The method of claim ~~[[25]]~~ 31, further comprising:  
storing in a control field associated with the current instruction a value to indicate sign  
extension compression.

Claim 27 (cancel)

Claim 28 (currently amended): The method of claim ~~[[27]]~~ 33, further comprising:  
storing in a control field associated with the next instruction a value to indicate that the  
compressed operand should be ignored when executing the next instruction.

Claim 29 (currently amended): The method of claim ~~[[22]]~~ 31, wherein  
compressing further comprises:  
compressing using sign extension.

Claim 30 (cancel)

Claim 31 (currently amended):     ~~The method of claim 30,~~ A method comprising:  
compressing an immediate operand associated with a current instruction;  
storing the compressed operand in a selected one of a plurality of fixed-length operand  
fields, wherein each of the operand fields is associated with one of a plurality of program  
instructions, the plurality of program instructions including the current instruction, a previous  
instruction and a next instruction; and

wherein storing the compressed operand further comprises:  
storing the compressed operand in the operand field for the previous instruction if the  
operand field of the previous instruction is available.

Claim 32 (currently amended):     The method of claim ~~[[30]]~~ 31, wherein storing the  
compressed operand further comprises:

storing the compressed operand in the operand field for the current instruction if the  
operand field of the previous instruction is unavailable and the operand field of the current  
instruction is available.

Claim 33 (currently amended):     The method of claim ~~[[30]]~~ 31, wherein storing the  
compressed operand further comprises:

storing the compressed operand in the operand field of the next instruction if the operand  
field of the previous instruction is unavailable and the operand field of the current instruction is  
unavailable and the operand field of the next instruction is available.

Claim 34 (currently amended):     The method of claim ~~[[30]]~~ 31, wherein storing the  
compressed operand further comprises:

generating a nop instruction if the operand field of the previous instruction is unavailable  
and the operand field of the current instruction is unavailable and the operand field of the next  
instruction is unavailable; and

storing the compressed operand in an operand field associated with the nop instruction.

Claim 35 (previously presented):     The method of claim 34, wherein generating the  
nop instruction further comprises:

inserting the nop instruction between the current instruction and the next instruction.

Claim 36 (cancel)

Claim 37 (currently amended): The method of claim ~~[[36]]~~ 39, wherein the length of the operand fields for the current instruction and the adjacent instruction is Y/2.

Claim 38 (canceled)

Claim 39 (currently amended): ~~The method of claim 36, further comprising: A~~  
method, comprising:

storing one portion of an immediate operand for a current instruction in a fixed-length operand field associated with the current instruction;

storing a remaining portion of the immediate operand for the current instruction in a fixed-length operand field of an instruction adjacent to the current instruction, wherein the adjacent instruction is a previous instruction;

wherein the length of the immediate operand is Y bits and the length of the operand fields for the current instruction and the adjacent instruction is less than Y bits; and

storing in a control field associated with the previous instruction a value to indicate backward scavenging compression.

Claim 40 (currently amended): The method of claim ~~[[36]]~~ 39, wherein the adjacent instruction is a next instruction.

Claim 41 (previously presented): The method of claim 40, further comprising:  
storing in a control field associated with the next instruction a value to indicate forward scavenging compression.

Claims 42 - 50 (canceled)

Claim 51 (currently amended): A method comprising:

compressing an immediate operand associated with a current instruction;  
storing the compressed immediate operand in one of a plurality of operand fields each associated with one of a plurality of instructions including the current instruction and a previous instruction; and  
storing the compressed immediate operand in the operand field for the previous instruction if the operand field of the previous instruction is available; and  
storing in a control field associated with the previous instruction a value to indicate that the compressed immediate operand should be ignored when executing the previous instruction.

Claim 52 (previously presented): The method of claim 51, further comprising storing the compressed immediate operand in the operand field for the current instruction if the operand field of the previous instruction is unavailable and the operand field of the current instruction is available.

Claim 53 (cancel)